Entrez Direct и NCBI Datasets CLI

и EMBOSS

EMBOSS

EMBOSS

European **M**olecular **B**iology **O**pen **S**oftware **S**uite

Пакет консольных биоинформатических программ.

- унифицированный интерфейс командной строки
- общий формат для задания адреса последовательностей (USA)
- есть программы для повседневных задач, возникающих при работе с биологическими последовательностями
- пакет перестал развиваться в 2013, программы устаревают

Помощь по программам

Можно получить справку в командной строке:

```
Краткое описание основных опций:
kodomo:~$ any-emboss-util -help

Описание всех имеющихся опций:
kodomo:~$ any-emboss-util -help -verbose

Подробное описание программы:
kodomo:~$ tfm 'any-emboss-util'

Поиск программы по описанию:
kodomo:~$ wossname 'alignment'

У всех программ есть man, по объему это примерно -help
kodomo:~$ man 'any-emboss-util'
```

Или можно читать описания в интернете:

http://emboss.open-bio.org/ официальный сайт, проблемы с доступом из-за РКН

https://emboss.sourceforge.net/ альтернативный сайт, лучше организован

Унифицированный адрес последовательности (USA)

Uniform **S**equence **A**ddress

Все варианты USA:

https://emboss.sourceforge.net/docs/themes/UniformSequenceAddress.html

Примеры (жирным выделены обязательные части):

Запись entry из базы данных dbname (аббревиатуры баз выдает `showdb`):

`dbname:entry``[start:end:reverse]`

Запись entry из файла file в формате format (названия форматов здесь):

`format::`**`file`**`:entry[start:end:reverse]`

Использовать все USA из файла listfile:

`@listfile`

В именах файлов и записей можно использовать маски (только `*` и `?`).
Не забывайте про экранирование!

Аргументы командной строки

- аргументы называются qualifiers
- бывают пяти типов: standard, additional, advanced, associated и general
- всегда задаются в виде опций, начинающихся с *одного* символа -
- название опции можно сокращать, пока сокращение уникально
- почти все опции требуют один аргумент
- у опций типа boolean аргумент можно опускать, имея в виду значение Y

Standard qualifiers

Обязательные аргументы

- если не заданы, будут запрошены с `STDIN` в процессе исполнения
- иногда могут задаваться в виде позиционных аргументов (т.е. без указания названия опции), если название опции в [] на странице `-help`
- иногда для них есть значение по умолчанию, которое можно активировать опцией `-auto`

```
Пример:
kodomo:~$ infoseq -sequence 'seq.fasta'
или (то же самое):
kodomo:~$ infoseq 'seq.fasta'
```

Additional qualifiers

Дополнительные аргументы

- если не заданы, будут использованы значения по умолчанию (но будут запрошены с `STDIN`, если задана опция `-options`)
- значения по умолчанию указаны в [] на странице `-help`

Пример:

```
kodomo:~$ infoseq seq.fasta -outfile 'report.txt'
```

Advanced qualifiers

Расширенные аргументы

- предполагается, что они редко потребуются рядовым пользователям
- отображаются на странице `-help` без опции `-verbose`

Пример:

```
kodomo:~$ infoseq seq.fasta -delimiter ';'
```

Associated qualifiers

Ассоциированные аргументы

- уточняют значения других аргументов
- не отображаются на странице `-help` без опции `-verbose`
- на странице `-help -verbose` указано, какой аргумент они уточняют

Пример:

```
kodomo:~$ infoseq seq.fasta -squick 'Y'
```

General qualifiers

Общие аргументы

- есть у всех программ EMBOSS
- не отображаются на странице `-help` без опции `-verbose` (за исключением самой опции `-help`)
- служат либо для получения служебной информации о программе, либо для переключения режима взаимодействия с программой

Пример:

```
kodomo:~$ infoseq -help 'Y' -verbose 'N'
```

Использование в конвейерах

Программы пакета EMBOSS неудобно использовать в конвейерах, ведь они:

- переключаются в интерактивный режим в случае указания не всех обязательных аргументов (даже при наличии подходящих умолчательных значений);
- по умолчанию используют файловый ввод/вывод (а не стандартные потоки);
- каждый раз выводят свое однострочное описание.

Использование в конвейерах

Есть общие (general) опции, позволяющие решить проблемы:

- `-auto` – использовать умолчательные значения для всех пропущенных аргументов (+ не выводить краткое описание);
- `-filter` – то же самое + заменить *умолчательные* ввод и вывод на стандартные потоки.

Опция `-filter` переключает потоки, только если ввод/вывод не указаны в аргументах явно. Советую *всегда* использовать эту опцию, не могу придумать ситуацию, когда она помешает.

В USA можно использовать специальные имена файлов `stdin` и `stdout`.

```
Изменить формат выдачи в терминал:
kodomo:~$ seqret 'seqs.fasta' 'plain::stdout'

Транслировать последовательность из потока ввода:
kodomo:~$ echo 'ATGC' | seqret -filter 'plain::stdin[::r]'
```

Проблемы с выводом сообщений

Программы EMBOSS выводят справку в `STDERR`, а не в `STDOUT`.

```
Слить STDOUT и STDERR и перенаправить в файл:
kodomo:~$ seqret -help &> 'seqret_help.txt'

Слить STDOUT и STDERR и передать следующей команде:
kodomo:~$ seqret -help -verbose |& less
```

В начале работы программы выводят свое краткое описание в `STDERR`.

```
Отключить сообщения на уровне команды EMBOSS:
kodomo:~$ seqret -filter 'seqs.fasta' | less
или
kodomo:~$ seqret -auto 'seqs.fasta' 'out.fasta'

Убить STDERR (перенаправить в черную дыру):
kodomo:~$ seqret 'seqs.fasta' 'plain::stdout' 2> '/dev/null' | less
```

Программа seqret

от **seq**uence **ret**urn

Получает последовательности по USA из первого аргумента и записывает их по USA из второго аргумента (можно указать только имя файла и его формат).

```
Изменить формат файла:
kodomo:~$ seqret -filter 'seqs.fasta' 'asn1::seqs.txt'

Вырезать участок последовательности:
kodomo:~$ seqret -filter 'seqs.fasta:*[1:1]' 'first_aa.fasta'

Отобрать последовательности по маске имени:
kodomo:~$ seqret -filter 'seqs.fasta:abc*' 'ncbi::abc.txt'

Объединить разные последовательности в один файл:
kodomo:~$ cat usa.list
seqs.fasta:qwe123[1:30]
seqs.fasta:qwe123[31:60]
ncbi::abc.txt
kodomo:~$ seqret -filter '@usa.list' 'ncbi::all.fasta'
```

Программа entret

от **ent**ry **ret**urn

Нужна для скачивания записей из баз данных.

```
Получить последовательность из базы можно с помощью seqret:
kodomo:~$ seqret -filter 'sw:ENO_ECOLI' eno_ecoli.fasta

Можно даже сохранить её в нативном формате:
kodomo:~$ seqret -filter 'sw:ENO_ECOLI' 'swiss::eno_ecoli.seqret'

Но seqret не гарантирует сохранение всех аннотаций!
kodomo:~$ grep -c '^FT' 'eno_ecoli.seqret'
0

Скачивать полные записи следует с помощью entret:
kodomo:~$ entret -filter 'sw:ENO_ECOLI' 'eno_ecoli.entret'
kodomo:~$ grep -c '^FT' 'eno_ecoli.entret'
151
```

Второй аргумент entret – **не** USA, а просто имя файла.

Другие полезные программы

infoseq – получить базовую информацию про последовательности в виде таблицы;

infoalign – сравнить последовательности в выравнивании с консенсусом;

getorf – найти открытые рамки считывания в нуклеотидной последовательности;

transeq – транслировать нуклеотидные последовательности;

wordcount – посчитать количество вхождений для k-меров заданной длины;

shuffleseq – перемешать буквы в последовательностях с сохранением состава;

makenucseq – сгенерировать случайные нуклеотидные последовательности;

makeprotseq – аналогично для белков;

... – программы выравниваний, поиска паттернов, работы с аннотациями и т.д.

Полный список программ с разбивкой по категориям:

```
kodomo:~$ wossname -filter | less
```

Полный список программ в алфавитном порядке:

```
kodomo:~$ wossname -filter -alphabetic | less
```

Entrez Direct (EDirect)

Entrez

Единая поисковая система, которая объединяет многие базы данных NCBI.

- У каждой базы данных в системе Entrez есть **имя**, оно не всегда совпадает с именем на сайте.

  Пример: nuccore – название в базы NCBI Nucleotide.
- Для каждой базы определены **поля** записей, по которым можно производить поиск.

  Пример: TIAB – поле Title/Abstract в PubMed.
- Между записями в базах данных есть **ссылки**, каждому типу ссылок присвоено свое имя.

  Пример: pubmed_pubmed_refs – ссылка из статьи в PubMed на цитируемую статью.

Функции Entrez

- Поиск в одной из баз данных NCBI – получение списка записей базы, удовлетворяющих поисковому запросу.
- Переход по ссылкам между записями – получение списка записей в той же или другой базе, которые связаны с указанными записями.
- Фильтрация списка записей согласно каким-либо критериям.
- Загрузка записей с возможностью выбора формата.

Все это вы можете делать на сайте NCBI.

Entrez API

У системы Entrez есть API, который позволяет использовать возможности Entrez в скриптах.

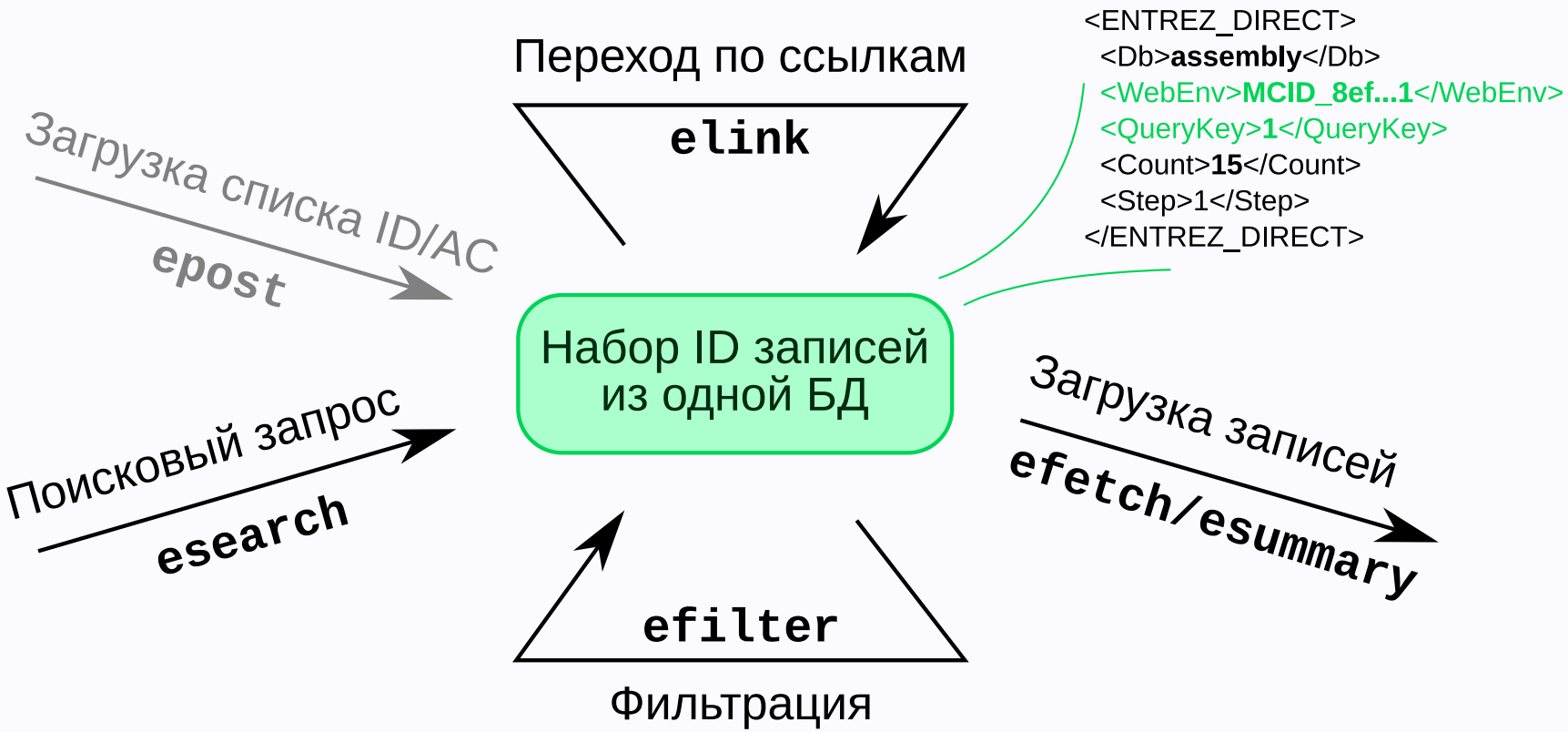
E-utilities	основной веб-API, все остальное работает через него. **Не советую использовать напрямую!**
Entrez Direct	набор консольных утилит для Unix-подобных систем, установлены на kodomo.
Bio.Entrez	модуль Biopython для работы с E-utilities.
…	

Entrez Direct (EDirect)

Набор консольных утилит:

einfo	получение списков названий баз данных, полей и ссылок
esearch	поисковые запросы к системе Entrez
elink	получение записей по ссылкам из других записей
efetch	скачивание записей по идентификаторам
esummary	получение основных полей записей
efilter	фильтрация результатов поиска
xtract	извлечение отдельных полей из выдачи в формате XML
transmute	изменение формата выдачи и не только
...	остальные нужны очень редко

EDirect: общая схема



Entrez History и конвейеры

Entrez хранит историю результатов выполнения запросов.

- Каждому запросу присваивается идентификатор WebEnv, по которому можно получить найденные записи.
- Утилиты EDirect могут работать напрямую с Entrez History, что позволяет производить операции с записями без загрузки их на локальный компьютер.
- Такая система позволяет объединять вызов утилит в конвейеры: между программами передается идентификатор запроса (и некоторая сопутствующая информация в формате XML), а операции с записями происходят на серверах NCBI.

Entrez History и конвейеры

Пример: поиск таксона по общепринятому названию, получение ссылок на геномные сборки и загрузка их идентификаторов:

```
$ esearch -db 'taxonomy' -query '"White shark" [COMN]' \
    | elink -target 'assembly' \
    | efetch -format 'uid'
9678721
9678001
2022931
```

EInfo

Самая важная программа, если вы не знаете всю структуру Entrez наизусть!

Получение информации о базах данных в системе Entrez:

```
Список баз данных
$ einfo -dbs

Список полей в базе данных
$ einfo -db 'taxonomy' -fields

Список названий ссылок на другие базы данных
$ einfo -db 'taxonomy' -links

Вся доступная информация про базу в формате JSON
$ einfo -db 'taxonomy' | transmute -x2j
```

EFetch и ESummary

Загрузка записей из базы данных:

```
По списку ID (или AC, раньше AC понимал только epost)
$ efetch -db 'protein' -id 'AAC74937.2,BAA15678.1' -format 'fasta'

Или так (если список ID хранится в файле ids.list)
$ efetch -db 'protein' -input 'ids.list' -format 'fasta'

Записи по идентификатору запроса (в виде WebEnv-XML на STDIN)
$ esearch/efilter/elink/epost ... | efetch -format 'docsum'

Список форматов зависит от базы
$ efetch -help # список неполный
```

ESummary – это синоним EFetch с опцией `-format docsum`.

ESearch

Поисковые запросы к системе Entrez:

```
В теории понимает полный синтаксис запросов Entrez:
$ esearch -query 'HhaI [TITL] AND Roberts* [AUTH] AND NAR [JOUR]' \
    -db 'pubmed' | efetch -format 'uid'
31879785
9207024
7753630
7899082
8506140
```

EFilter

Фильтрация записей по дополнительным критериям:

```
Опция -query аналогична esearch:
$ esearch -query 'HhaI [TITL] AND Roberts* [AUTH]' -db 'pubmed' \
    | efilter -query 'NAR [JOUR]' | efetch -format 'uid'
31879785
9207024
7753630
7899082
8506140
```

ELink

Работа с перекрестными ссылками:

```
Переход по ссылкам на другие записи
$ elink -db 'protein' -id 'AAC74937.2,CAB15395.1' \
    -target 'nuccore' | esummary -json
 ... загрузка нуклеотидных записей

Получение ссылок на записи в других базах данных
$ elink -db 'protein' -id 'AAC74937.2,CAB15395.1' \
    -target 'taxonomy' -cmd 'neighbor' \
    | xtract -pattern 'LinkSet' -element 'IdList/Id,Link/Id'
 ... выдача пар белок-taxid
```

Справка по EDirect

Нужно заранее знать, где искать информацию требуемого типа!

Описание опций программ:

```
У программ есть справочные страницы в системе man
$ man esearch
Встроенная помощь иногда неполная, зато часто содержит примеры
$ elink -help | less
Встроенная помощь по efetch содержит список возможных форматов для многих баз
$ efetch -help | less '+/^Format Examples'
```

Названия баз/полей/ссылок – только через einfo:

```
Список названий баз
$ einfo -dbs
Подробная информация про базу, включая список полей и типов ссылок
$ einfo -db 'nuccore' | transmute -x2j
```

Синтаксис поисковых запросов – только в справке по Entrez на сайте NCBI:

Entrez https://www.ncbi.nlm.nih.gov/books/NBK3837

Про многое подробнее можно почитать в руководствах на сайте:

EDirect https://www.ncbi.nlm.nih.gov/books/NBK179288

E-utilities https://www.ncbi.nlm.nih.gov/books/NBK25501

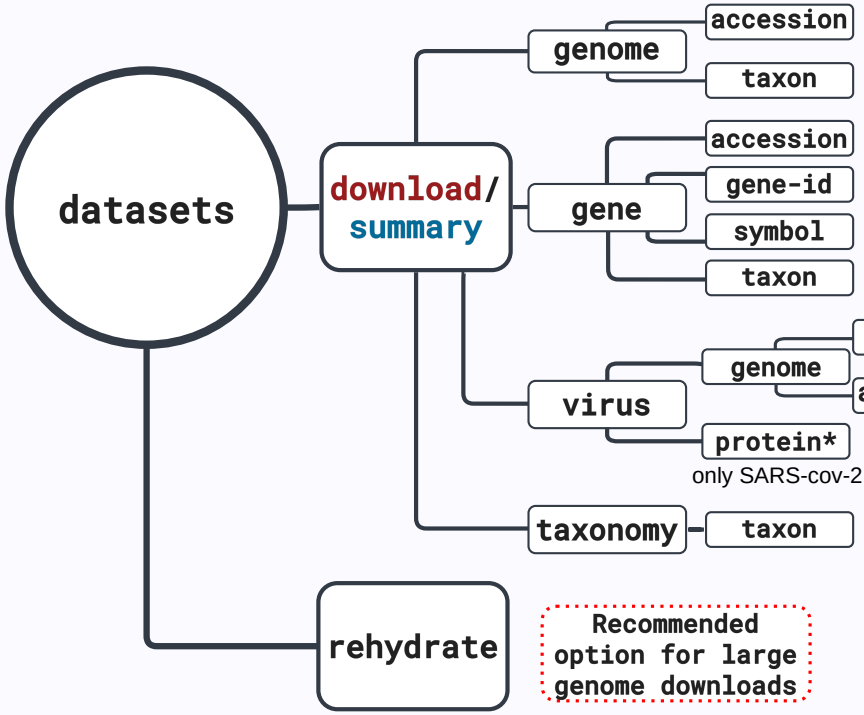
NCBI Datasets CLI

NCBI Datasets

Ребрендинг геномных данных в NCBI.

- БД Assembly переименовали в Datasets/Genome.
- БД Genome переименовали в Datasets/Taxonomy и существенно доработали.
- Улучшили веб-интерфейс обеих баз, добавили удобный поиск и навигацию.
- Создали новое независимое CLI API – консольные программы `datasets` и `dataformat`.

datasets: примеры команд




```
datasets download genome accession GCF_000001405.40
datasets summary  genome taxon ailuridae
```

```
datasets download gene accession NP_000483.3
datasets download gene symbol brca1 --ortholog primates
datasets summary  gene gene-id 40650
```

```
datasets download virus genome taxon monkeypox
datasets summary  virus genome accession NC_045521.2
datasets download virus protein ORF10
```

```
datasets download taxonomy taxon "mus musculus"
datasets summary  taxonomy taxon mustelidae --children
```

```
datasets rehydrate --directory my_genomes
```

https://www.ncbi.nlm.nih.gov/datasets/docs/v2/datasets_schema_taxonomy.svg

datasets: используемые форматы

JSON – умолчательный формат для `summary`, можно парсить только сторонними программами (например, `jq`) или самописными скриптами на Python (модуль `json`).

JSON Lines – формат выдачи `summary` с опцией `-as-json-lines` – набор строк, каждая является объектом JSON, можно перевести в таблицу TSV или XLSX с помощью `dataformat`.

ZIP – `download` создает zip-архив, формат файлов в архиве фиксированный, но можно изменить их список; для распаковки архива используйте `unzip`.

Программа `dataformat`:

- позволяет превращать JSON Lines в таблицы;
- не понимает произвольные JSON Lines, только выдачу `datasets`;
- не понимает обычный JSON (даже выдачу `datasets`).